Listing of Claims

Claim 1 (Currently Amended): A method of accessing data contained in a first file, wherein said first file is comprised in a plurality of files stored on a secondary storage, said secondary storage comprising a plurality of clusters, wherein a cluster is a basic unit of allocation for storing data related to a file and is identified on said secondary storage by a corresponding one of a plurality of identifiers, a file allocation table (FAT) indicating a corresponding set of clusters allocated to each of said plurality of files by associating the corresponding set of identifiers with each file whereby said FAT indicates that a first set of clusters are allocated to said first file by associating a first set of identifiers with said first file, each of said first set of identifiers identifying that a corresponding one of said first set of clusters is allocated to said first file, said first set of identifiers being contained in said plurality of identifiers, said FAT being stored on said secondary memory, said method being performed in a single device, said method comprising:

determining ~~a~~ said first set of identifiers by retrieving and examining said FAT stored in said secondary memory, ~~wherein each of said set of identifiers identifies a corresponding one of said set of clusters allocated to said first file~~;

storing said first set of identifiers associated with said first file in a random access memory (RAM) indicating that said first set of clusters store data related to said file; and

retrieving at least a portion of said first file from said secondary storage based on said first set of identifiers stored in said RAM,

wherein said determining and said storing are performed when the content of said first file is to be retrieved from said secondary storage for processing.

Claim 2 (Currently Amended): The method of claim 1, ~~wherein each of said plurality of clusters is identified by a corresponding one of a plurality of identifiers,~~ said FAT storing

said <u>first</u> set of identifiers in the form of a linked list, wherein an order specified by said

linked list indicates the sequence in which said set of clusters are used to store data contained

in said first file, said method comprising:

traversing said linked list to retrieve said <u>first</u> set of identifiers in said order, wherein

said storing stores said <u>first</u> set of identifiers in said RAM.

Claim 3 (Currently Amended): The method of claim 2, wherein said <u>first</u> set of

identifiers are stored according to a technique which permits each of said <u>first</u> set of

identifiers to be retrieved with fewer instructions than the number of instructions required to

access the same identifier from said FAT in said secondary storage.

Claim 4 (Currently Amended): The method of claim 2, wherein said <u>first</u> set of

identifiers are stored in the form of an array <u>in said RAM</u> which permits each identifier to be

retrieved by a single access.

Claim 5 (Currently Amended): The method of claim 4, further comprising:

receiving a start offset of data to be accessed;

computing a cluster index by dividing said start offset by a number of bytes in each

of said plurality of clusters; and

accessing said array using said cluster index to determine a specific one of said <u>first</u>

set of identifiers, wherein said data to be accessed is present in a cluster identified by said

specific one of said <u>first</u> set of identifiers.

Claim 6 (Currently Amended): The method of claim 5, wherein data stored in said first file represents a song and wherein said single device is designed to play said song based on the stored data, said method further comprising:

receiving a request for rewind operation requiring access to data in a previous cluster when playing said song;

accessing the data in said previous cluster using said first set of identifiers stored in said RAM .

Claim 7 (Currently Amended): A method of implementing an application in a system containing a small memory in the form of a random access memory (RAM), wherein said system supports a file system on a secondary storage, wherein said secondary storage comprises a plurality of clusters each identified by a corresponding identifier in said secondary storage, wherein said file system comprises a plurality of files and each of said plurality of files is stored in a corresponding one of a plurality of sets of clusters, a file allocation table (FAT) stored on said secondary storage indicating a corresponding set of clusters allocated to each of said plurality of files by associating corresponding set of identifiers with each file, said plurality of sets of clusters being contained in said plurality of clusters, said method comprising:

providing a first module which is designed to determine a plurality of identifiers corresponding to a specified file by examining said FAT and store said plurality of identifiers in a random access memory (RAM) said RAM according to a convention, wherein said plurality of identifiers specify a set of clusters corresponding to said specified file, said set of clusters being contained in said plurality of sets of clusters;

providing a second module which is to perform an operation on a file of interest,

wherein said second module is designed to determine a desired cluster by using said plurality

of identifiers stored in said RAM according to said convention;

executing said first module when the content of said specified file is to be retrieved

from said secondary storage for processing such that only a portion of said FAT including

data indicating that said plurality of identifiers identify the clusters storing data related to said

specified file is stored in said small memory,

_____wherein said first module is executed while specifying said file of interest as said

specified file such that said plurality of identifiers corresponding to said file of interest are

stored in said RAM according to said convention; and

executing said second module after executing said first module,

wherein both of said first module and said second module are executed using at least

some of the same locations of said small memory.

Claim 8 (Original): The method of claim 7, wherein said second module is overlaid

on the same memory space on which said first module is loaded during execution.

Claim 9 (Previously Presented): The method of claim 8, wherein said convention

comprises storing said plurality of identifiers at a pre-specified portion of said RAM.

Claim 10 (Original): The method of claim 9, wherein each of said plurality of files

stores data representing a corresponding song.

Claim 11 (Currently Amended): A machine readable medium carrying one or more

sequences of instructions for causing a single digital processing system to access data

contained in a first file, wherein said first file is comprised in a plurality of files stored on a

secondary storage, said secondary storage comprising a plurality of clusters each identified

on said secondary storage by a corresponding one of a plurality of identifiers, a file allocation

table (FAT) indicating a corresponding set of clusters allocated to each of said plurality of

files by associating the corresponding set of identifiers with each file whereby said FAT

indicates that a first set of clusters are allocated to said first file by associating a first set of

identifiers with said first file, said first set of identifiers being contained in said plurality of

identifiers, said FAT being stored on said secondary memory, wherein execution of said one

or more sequences of instructions by one or more processors contained in said digital

processing system causes said one or more processors to perform the actions of:

determining ~~a~~ said first set of identifiers by retrieving and examining said FAT stored

in said secondary memory, ~~wherein each of said set of identifiers identifies a corresponding~~

~~one of said set of clusters allocated to said first file~~;

storing said first set of identifiers associated with said first file in a random access

memory (RAM) indicating that said first set of clusters store data related to said file; and

retrieving at least a portion of said first file from said secondary storage based on said

first set of identifiers stored in said RAM,

wherein said determining and said storing are performed when the content of said first

file is to be retrieved from said secondary storage for processing.

Claim 12 (Currently Amended): The machine readable medium of claim 11, wherein

each of said plurality of clusters is identified by a corresponding one of a plurality of

identifiers, said FAT storing said first set of identifiers in the form of a linked list, wherein

an order specified by said linked list indicates the sequence in which said set of clusters are

used to store data contained in said first file, further comprising:

traversing said linked list to retrieve said <u>first</u> set of identifiers in said order, wherein said storing stores said <u>first</u> set of identifiers in said RAM.

Claim 13 (Currently Amended): The machine readable medium of claim 12, wherein said <u>first</u> set of identifiers are stored according to a technique which permits each of said <u>first</u> set of identifiers to be retrieved with fewer instructions than the number of instructions required to access the same identifier from said FAT in said secondary storage.

Claim 14 (Currently Amended): The machine readable medium of claim 12, wherein said <u>first</u> set of identifiers are stored in the form of an array <u>in said RAM</u> which permits each identifier to be retrieved by a single access

Claim 15 (Canceled)

Claim 16 (Currently Amended): The machine readable medium of claim 14, further comprising:

receiving a start offset of data to be accessed;

computing a cluster index by dividing said start offset by a number of bytes in each of said plurality of clusters; and

accessing said array using said cluster index to determine a specific one of said <u>first</u> set of identifiers, wherein said data to be accessed is present in a cluster identified by said specific one of said <u>first</u> set of identifiers.

Claim 17 (Currently Amended): A machine readable medium carrying one or more sequences of instructions for causing a digital processing system to implement an application

using a small memory space in a random access memory (RAM), wherein said digital

processing system supports a file system on a secondary storage, wherein said secondary

storage comprises a plurality of clusters each identified by a corresponding identifier in said

secondary storage, wherein said file system comprises a plurality of files and each of said

plurality of files is stored in a corresponding one of a plurality of sets of clusters, said

plurality of sets of clusters being contained in said plurality of clusters, a file allocation table

(FAT) stored in said secondary storage indicating a corresponding set of clusters allocated

to each of said plurality of files, wherein execution of said one or more sequences of

instructions by one or more processors contained in said digital processing system causes said

one or more processors to perform the actions of:

     providing a first module which is designed to determine a plurality of identifiers

corresponding to a specified file by examining said FAT and store said plurality of identifiers

in a random access memory (RAM) said RAM according to a convention, wherein said

plurality of identifiers specify a set of clusters corresponding to said specified file, said set

of clusters being contained in said plurality of sets of clusters;

     providing a second module which is to perform an operation on a file of interest,

wherein said second module is designed to determine a desired cluster by using said plurality

of identifiers stored in said RAM according to said convention;

     executing said first module when the content of said specified file is to be retrieved

from said secondary storage for processing such that only a portion of said FAT including

data indicating that said plurality of identifiers identify the clusters storing data related to said

specified file is stored in said small memory,

     wherein said first module is executed while specifying said file of interest as said

specified file such that a plurality of identifiers corresponding to said file of interest are stored

in said RAM according to said convention; and

executing said second module after executing said first module,

wherein both of said first module and said second module are executed using at least

some of the same locations of said small memory.

Claim 18 (Original):The machine readable medium of claim 17, wherein said second

module is overlaid on the same memory space on which said first module is loaded during

execution.

Claim 19 (Previously Presented):The machine readable medium of claim 18, wherein

said convention comprises storing said plurality of identifiers at a pre-specified portion of

said RAM.

Claim 20 (Currently Amended): An apparatus accessing data contained in a first file,

wherein said first file is comprised in a plurality of files stored on a secondary storage, said

secondary storage comprising a plurality of clusters, wherein a cluster is a basic unit of

allocation for storing data related to a file and is identified on said secondary storage by a

corresponding one of a plurality of identifiers, a file allocation table (FAT) indicating a

corresponding set of clusters allocated to each of said plurality of files by associating the

corresponding set of identifiers with each file whereby said FAT indicates that a first set of

clusters are allocated to said first file by associating a first set of identifiers with said first file,

each of said first set of identifiers identifying that a corresponding one of said first set of

clusters is allocated to said first file, said first set of identifiers being contained in said

plurality of identifiers, said FAT being stored on said secondary memory, said apparatus

comprising:

means for determining ~~a~~ <u>said first</u> set of identifiers by <u>retrieving and</u> examining said

FAT <u>stored in said secondary memory</u>, ~~wherein each of said set of identifiers identifies a~~

~~corresponding one of said set of clusters allocated to said first file~~;

means for storing said <u>first</u> set of identifiers <u>associated with said first file</u> in a random

access memory (RAM) <u>indicating that said first set of clusters store data related to said file</u>;

and

means for retrieving at least a portion of said first file from said secondary storage

based on said <u>first</u> set of identifiers stored in said RAM,

wherein said determining and said storing are performed when the content of said first

file is to be retrieved from said secondary storage for processing.

Claim 21 (Currently Amended): The apparatus of claim 20, ~~wherein each of said~~

~~plurality of clusters is identified by a corresponding one of a plurality of identifiers~~, said FAT

storing said <u>first</u> set of identifiers in the form of a linked list, wherein an order specified by

said linked list indicates the sequence in which said set of clusters are used to store data

contained in said first file, said apparatus further comprising:

means for traversing said linked list to retrieve said <u>first</u> set of identifiers in said order,

wherein said storing stores said <u>first</u> set of identifiers in said RAM.

Claim 22 (Currently Amended): The apparatus of claim 21, wherein said <u>first</u> set of

identifiers are stored according to a technique which permits each of said <u>first</u> set of

identifiers to be retrieved with fewer instructions than the number of instructions required to

access the same identifier from said FAT in said secondary storage.

Claim 23 (Currently Amended): The apparatus of claim 21,wherein said first set of

identifiers are stored in the form of an array in said RAM which permits each identifier to be

retrieved by a single access.

Claim 24 (Canceled)

Claim 25 (Currently Amended): The apparatus of claim 24, further comprising:

means for receiving a start offset of data to be accessed;

means for computing a cluster index by dividing said start offset by a number of bytes

in each of said plurality of clusters; and

means for accessing said array using said cluster index to determine a specific one of

said first set of identifiers, wherein said data to be accessed is present in a cluster identified

by said specific one of said first set of identifiers.

Claims 26 - 28 (Canceled)